Unconstrained and Constrained Global Optimization of Polynomial Functions in One Variable

V. Visweswaran[1] C.A. Floudas[1 2]

Abstract

In Floudas and Visweswaran (1990), a new global optimization algorithm **(GOP)** was proposed for solving constrained nonconvex problems involving quadratic and polynomial functions in the objective function and/or constraints. In this paper, the application of this algorithm to the special case of polynomial functions of one variable is discussed. The special nature of polynomial functions enables considerable simplification of the **GOP** algorithm. The primal problem is shown to reduce to a simple function evaluation, while the *relaxed dual* problem is equivalent to the simultaneous solution of two linear equations in two variables. In addition, the one-to-one correspondence between the x and y variables in the problem enables the iterative improvement of the bounds used in the relaxed dual problem. The simplified approach is illustrated through a simple example that shows the significant improvement in the underestimating function obtained from the application of the modified algorithm. The application of the algorithm to several unconstrained and constrained polynomial function problems is demonstrated.

Keywords : Global Optimization, Polynomial functions, Unconstrained and Constrained Optimization, The GOP algorithm.

1 Introduction

Polynomial functions of one variable occur frequently in mathematical programming problems. Problems involving the unconstrained or constrained optimization of these functions are interesting not only because of the inherent simplicity of the problem structure, but also because these functions form the backbone of larger optimization problems involving more variables. Often, the solution of these larger problems becomes much easier if a few of the variables are fixed. Consequently, they can be viewed as parametric problems in one variable. The solution of optimization problems involving one (or a few) variable(s) can often provide significant insight into the nature of larger problems.

The unconstrained minimization of Lipschitz continuous functions (of which polynomial functions are a subset) has been studied extensively

[1] Chemical Engineering Department, Princeton University, Princeton, N.J. 08544
[2] Author to whom all correspondence should be addressed.

in the past two decades. Algorithms for solving this problem have been proposed by Evtushenko (1971), Piyavskii (1972), and Timonov (1977), among others. Shen and Zhu (1987) proposed an interval version of Schubert's algorithm for univariate functions. Galperin (1987) and Pinter(1988) also considered the incorporation of constraints in the problem. Hansen (1979) proposed an algorithm for minimizing univariate functions using interval analysis. A comprehensive review of global optimization of univariate Lipschitz functions (including functions other than polynomials) is given in Hansen *et al* (1989a). They provide the necessary conditions for finite convergence of algorithms addressing this problem and the characteristic that a best possible algorithm should have. An extensive comparison of the computational aspects of these algorithms as well as new improved algorithms are provided in Hansen *et al* (1989b). Wingo (1985) proposed a method for locally approximating the polynomial function to enable the solution without evaluating derivatives. However, the algorithm fails to identify the global solution in some cases. Dixon (1990) proposed several methods for accelerating the search procedure using interval methods to locate the global solution for functions of one variable.

Floudas and Visweswaran (1990) proposed a deterministic approach for global optimization of problems involving quadratic and polynomial functions in the objective function and/or constraints. They made use of primal-dual decomposition to solve the originally nonconvex problem through a series of primal and *relaxed dual* subproblems. The algorithm **(GOP)** was shown to have finite convergence to an ϵ-global minimum of the problem. The algorithm was applied to several classes of problems including polynomial function problems. Visweswaran and Floudas (1990b) presented new properties that exploit the structure of the Lagrange function and showed that they enhance the computational efficiency of the **GOP** algorithm when applied to problems with quadratic terms in the objective function and/or constraints.

In this paper, the application of the **GOP** algorithm to the special case of polynomial functions in one variable is discussed. Use is made of the new properties presented in Visweswaran and Floudas (1990b) as well as the structure of polynomial functions to reduce the **GOP** algorithm to an extremely simple form of application. The modified algorithm is also shown to be applicable to constrained optimization problems with polynomial functions in one variable. The improvements over the original **GOP** algorithm are illustrated both computationally and geometrically through the use of a simple example. In addition, several examples of unconstrained and constrained problems help to highlight the effectiveness of the proposed algorithm.

2 Problem Statement

In this paper, the application of the **GOP** algorithm to optimization problems involving polynomial functions of one variable in the objective function and/or constraints is presented. These problems have the following form :

$$\min \quad F(y) = a_0 + a_1 y + a_2 y^2 + ... + a_N y^N$$

$$\begin{aligned} A_{j0} + A_{j1} y + A_{j2} y^2 + ... + A_{jN} y^N &\leq 0 \quad \forall\, j = 1,2,...,J \\ B_{m0} + B_{m1} y + B_{m2} y^2 + ... + B_{mN} y^N &= 0 \quad \forall\, m = 1,2,...,M \\ y^L \leq y \leq y^U \ , \end{aligned} \tag{1}$$

where y is a *single* variable and A_{ji}, A_{mi} are the coefficients of y^i in the jth inequality and mth equality constraint respectively. The nonconvexities in this problem arise due to the existence of polynomial terms in either the objective function or the set of constraints. It is assumed that the polynomials have nonconvex terms right up to the N^{th} degree term.

Consider the following transformations:

$$\begin{aligned} x_0 &= 1 \\ x_1 &= y \\ x_2 &= y^2 = (x_1)y \\ x_3 &= y^3 = (x_2)y \\ &\vdots \\ x_N &= y^N = (x_{N-1})y. \end{aligned}$$

Then, the problem can be written in the following equivalent form:

$$\min_{x,y} \sum_{i=0}^{N} a_i x_i$$

$$\begin{aligned} \sum_{i=0}^{N} A_{ji} x_i &\leq 0 \quad j = 1,2,...,J \\ \sum_{i=0}^{N} B_{mi} x_i &= 0 \quad m = 1,2,...,M \\ x_i - x_{i-1} y &= 0 \quad i = 1,2,...,N \end{aligned} \tag{2}$$

where $x_0 = 1$.

The case of unconstrained optimization problems is considered in the following section. The theoretical development is extended to the case of constrained problems in Section 7 .

3 Unconstrained Problems

For the case of unconstrained optimization, problem (2) can be simplified to the following form :

$$\min_{x,y} \sum_{i=0}^{N} a_i x_i$$

$$x_i - x_{i-1} y = 0 \quad i = 1, 2, ..., N, \tag{3}$$

where $x_0 = 1$. For a fixed $y = y^K$, the primal problem can be written as

$$\min_{x} \sum_{i=0}^{N} a_i x_i$$

$$s.t. \qquad x_i - x_{i-1} y^K = 0 \quad i = 1, 2, ..., N \tag{4}$$

with $x_0 = 1$.

Note that for any fixed value of $y = y^k$, all the x variables are uniquely determined. Therefore, this problem is simply one of function evaluation, with the solution being as follows :

$$\begin{array}{ccc} x_0 & = & 1 \\ x_1 & = & y^K \\ x_2 & = & (y^K)^2 \\ x_3 & = & (y^K)^3 \\ \vdots & & \vdots \\ x_N & = & (y^K)^N \\ F(y^K) & = & \sum_{i=0}^{N} a_i (y^K)^i. \end{array}$$

The KKT gradient conditions for the primal problem (4), for a fixed $y = y^K$, can be written as :

$$\nabla_{x_i} L(x, y^K, \nu^K) = a_i + \nu_i^K - \nu_{i+1}^K y^K = 0 \quad \forall\, i = 1, 2, ..., N$$

where ν_i^K are the Lagrange multipliers for the new equality constraints introduced, with $\nu_0^K = \nu_{N+1}^K = 0$. Here, the Lagrange multipliers can be found by backward substitution :

$$\begin{array}{lllll} \nu_N^K & = & -a_N & & \\ \nu_{N-1}^K & = & \nu_N^K y^K - a_{N-1} & = & -a_N(y^K) - a_{N-1} \\ \nu_{N-2}^K & = & \nu_{N-1}^K y^K - a_{N-2} & = & -a_N(y^K)^2 - a_{N-1}(y^K) - a_{N-2} \\ \vdots & & \vdots & & \vdots \\ \nu_1^K & = & \nu_2^K y^K - a_1^K & = & -a_N(y^K)^{N-1} \cdots - a_2(y^K) - a_1. \end{array}$$

The Lagrange function formulated from the primal problem (4) can be written as

$$L(x, y, \nu^K) = \sum_{i=0}^{N} a_i x_i + \sum_{i=1}^{N} \nu_i^K (x_i - x_{i-1} y).$$

Separating the terms in x, this can be rewritten as

$$L(x, y, \nu^K) = \sum_{i=0}^{N} \left[a_i + \nu_i^K - \nu_{i+1}^K y\right] x_i. \tag{5}$$

Using the KKT gradient conditions, the Lagrange function becomes

$$L(x, y, \nu^K) = \sum_{i=0}^{N} \nu_{i+1}^K (y^K - y) x_i.$$

Thus, the *qualifying* constraint for each x_i is of the form

$$y^K - y \geq 0, \;\; or \quad y^K - y \leq 0,$$

depending on whether ν_{i+1}^k is greater than or less than zero respectively. Therefore, two relaxed dual problems are solved at every iteration. These two problems are separately considered below.

First Relaxed Dual Problem

Consider the relaxed dual problem solved for the region $y^K - y \geq 0$. Before solving the relaxed dual problems, the Lagrange functions from previous iterations are selected. From the kth iteration ($k = 1, 2, ..., K - 1$), the two Lagrange functions have *qualifying* constraints of the form $y^k - y \geq 0$ and $y^k - y \leq 0$. These Lagrange functions are selected on the basis of satisfaction of their *qualifying* constraints at $y = y^K$. Since $y^k - y^K$ must be either positive or negative, exactly one Lagrange function from every iteration will be present for the current relaxed dual problems.

Now, the previous iterations correspond to values of $y = y^1, \ldots, y^{K-1}$. Some of these fixed values of y will be less than y^K. Suppose that

$$y^L = \max_{j=1,\ldots,K-1} \{y^j : y^j < y^K\}.$$

Then, the Lagrange function formulated from the Lth iteration (for which the fixed value of y is y^L) has the *qualifying* constraint $y^L - y \leq 0$. Therefore, for the current relaxed dual problem, the lower bound on the y variable is y^L. At the same time, the *qualifying* constraint from the current iteration ensures that the upper bound on y is y^K. Thus, for the current relaxed dual problem, $y^L \leq y \leq y^K$.

Now, the x variables are related to y through the equivalence relations

$$\begin{array}{lcccc} x_0 & = & 1 & & \\ x_1 & = & y & & \\ x_2 & = & y^2 & = & (x_1)y \\ \vdots & & \vdots & & \vdots \\ x_N & = & y^N & = & (x_{N-1})y. \end{array}$$

Therefore, the bounds on the x variables can be changed to suit the new bounds on y for the current relaxed dual problem. This is done in the following manner :

$$\begin{array}{lclclcl} x_1^L & = & y^L , & & x_1^U & = & y^K \\ x_2^L & = & MIN[(y^L)^2, (y^K)^2] , & & x_2^U & = & MAX[(y^L)^2, (y^K)^2] \\ x_3^L & = & MIN[(y^L)^3, (y^K)^3] , & & x_3^U & = & MAX[(y^L)^3, (y^K)^3] \\ \vdots & & \vdots & & \vdots & & \vdots \\ x_N^L & = & MIN[(y^L)^N, (y^K)^N] , & & x_N^U & = & MAX[(y^L)^N, (y^K)^N]. \end{array}$$

If $y^L < 0$ and $y^K > 0$, then $x_2^L = x_4^L = x_6^L \ldots = 0$.

The reason for the use of these expressions is as follows. If $y^L \geq 0$, then the lower and upper bounds on $x_1, x_2, \ldots x_N$ are simply given by

$$\begin{aligned} x^L &= (y^L, (y^L)^2, \ldots (y^L)^N), \text{ and} \\ x^U &= (y^K, (y^K)^2, \ldots (y^K)^N). \end{aligned}$$

However, if $y^L < 0$ and $y^K > 0$, then the value of y can be either positive or negative. Therefore, any even power of y can be as low as zero. Hence, $x_2^L = x_4^L \ldots = 0$. Furthermore, $\mid y^L \mid$ can be either greater than or less than $\mid y^K \mid$. Therefore, for each of the powers of y, the minimum and maximum values are given by the minimum and maximum values of the corresponding terms in *both* y^L and y^K.

These new bounds for the x variables are then used in the Lagrange functions in the following form :

$$\begin{array}{llll} If & \nu_{i+1} \geq 0, & then & x_i^B = x_i^L. \\ If & \nu_{i+1} < 0, & then & x_i^B = x_i^U. \end{array}$$

It should be noted here that due to the nature of the transformation variables that are introduced into the problem, the Lagrange function is linear in x for every fixed y. Therefore, there is no need to linearize the Lagrange function with respect to x (Visweswaran and Floudas, 1990).

Now, there are $(K - 1)$ Lagrange functions from previous iterations that can be used for the current relaxed dual problem. However, it is sufficient to consider only one of these Lagrange functions, since the omission of the remaining constraints does not destroy the validity of the relaxed dual problem as a lower bound on the global solution. The obvious choice for this is to use the Lagrange function corresponding to the nearest point on the left side, that is, the Lagrange function corresponding to y^L, and ignoring all other Lagrange functions from previous iterations. The current relaxed dual problem then has the following form :

$$\min_{y, \mu_B} \mu_B$$

s.t.

$$\begin{aligned} \mu_B \geq \quad & L(x^{B_L}, y, \nu^L) \\ & y \geq y^L \\ \mu_B \geq \quad & L(x^{B_K}, y, \nu^K) \\ & y \leq y^K \end{aligned}$$

where B_L and B_K are two combinations of bounds of the x variables that are being used in the Lagrange functions from the Lth and Kth iterations, respectively.

Again, due to the nature of the transformations, the two constraints are linear in y. Therefore, it is clear that the solution to this problem will be either at the intersection of the two constraints, or at one of the two bounds for y, namely, y^L or y^K. Consider the three cases.

(i) The solution of the current relaxed dual problem lies at y^L. In this case, the value of the first Lagrange function (that is, the one formulated from y^L) must equal the value of the objective function at y^L. This arises as a result of the strong duality theorem. This means that due to the presence of the first constraint, the value of μ_B must equal the value of the objective function at y^L. Therefore, it is not necessary to consider this solution.

(i) The solution of the current relaxed dual problem lies at y^K. In this case, the value of the second Lagrange function (that is, the one formulated from y^K) must equal the value of the objective function at y^K. This arises as a result of the strong duality theorem. This means that due to the presence of the second constraint, the value of μ_B must equal the value of the objective function at y^K. Therefore, again it is not necessary to consider this solution.

(iii) The solution of the current relaxed dual problem lies at the intersection of the two constraints. In this case, the solution cannot be omitted.

Thus, the relaxed dual problem can be solved easily by considering only the third possibility and solving for the intersection of the two constraints in the problem. This is very easy to do since the constraints are linear in y.

Second Relaxed Dual Problem

For the case when $y^K - y \leq 0$, the nearest point to the right side of y^K becomes the upper bound for y, while the value of y^K becomes the lower bound for y. That is , for this case, $y^K \leq y \leq y^R$, where now y^R is found as

$$y^R = \min_{j=1,\dots,K-1} \{y^j : y^j > y^K\} \; .$$

Similarly, the bounds for the x variables can be found as follows :

$$\begin{array}{lclclcl}
x_1^L & = & y^K \; , & & x_1^U & = & y^R \\
x_2^L & = & MIN[(y^K)^2, (y^R)^2] \; , & & x_2^U & = & MAX[(y^K)^2, (y^R)^2] \\
x_3^L & = & MIN[(y^K)^3, (y^R)^3] \; , & & x_3^U & = & MAX[(y^K)^3, (y^R)^3] \\
\vdots & & \vdots & & \vdots & & \vdots \\
x_N^L & = & MIN[(y^K)^N, (y^R)^N] \; , & & x_N^U & = & MAX[(y^K)^N, (y^R)^N] \; .
\end{array}$$

If $y^K < 0$, then $x_2^L = x_4^L = x_6^L \ldots = 0$.

In this case, the relaxed dual problem is solved by considering the current Lagrange function, that is, the Lagrange function formulated from the current iteration corresponding to $y^K - y \leq 0$, and the Lagrange function from the Rth iteration corresponding to $y^R - y \geq 0$. Again, the relaxed dual problem can be solved simply by considering the intersection of the two Lagrange functions and comparing the corresponding value of μ_B to the values of the objective function at y^K and y^R.

4 The Improved GOP Algorithm

The modified algorithm for minimizing unconstrained polynomial functions in one variable can be stated in the following steps :

STEP 0– *Initialization of parameters:*

Define the storage parameters $\mu_B^{stor^1}(K^{max})$, $\mu_B^{stor^2}(K^{max})$, $y^{stor^1}(K^{max})$, $y^{stor^2}(K^{max})$, and $y^k(K^{max})$ over the maximum expected number of iterations K^{max}. Define P^{UBD} and R^{LBD} as the upper and lower bounds obtained from the primal and relaxed dual problems respectively. Also define the parameters y^{LEFT} and y^{RIGHT} and initialize them to the original lower and upper bounds y^L, y^U on y. Set

$$\mu_B^{stor^1}(K^{max}) = U, \mu_B^{stor^2}(K^{max}) = U$$

$$P^{UBD} = U, \quad and \ R^{LBD} = L.$$

where U is a very large *positive* number and L is a very large *negative* number. Define the logical variables LRD and RRD (for the left and right relaxed dual problems, respectively). Select a starting point y^1 for the algorithm. Set the counter K equal to 1. Select a convergence tolerance parameter ϵ.

STEP 1– *Primal problem:*

Store the value of y^K. Calculate the solution of the problem as follows :

$$\begin{array}{rcl} x_0 & = & 1 \\ x_1 & = & (y^K) \\ x_2 & = & (y^K)^2 \\ x_3 & = & (y^K)^3 \\ \vdots & & \vdots \\ x_N & = & (y^K)^N \\ F(y^K) & = & \sum_{i=0}^{N} a_i (y^K)^i \ . \end{array}$$

Also find the Lagrange multipliers for the problem as follows :

$$\begin{array}{rclcl} \nu_N^K & = & -a_N & & \\ \nu_{N-1}^K & = & \nu_N^K y^K - a_{N-1} & = & -a_N(y^K) - a_{N-1} \\ \nu_{N-2}^K & = & \nu_{N-1}^K y^K - a_{N-2} & = & -a_N(y^K)^2 - a_{N-1}(y^K) - a_{N-2} \\ \vdots & & \vdots & & \vdots \\ \nu_1^K & = & \nu_2^K y^K - a_1^K & = & -a_N(y^K)^{N-1} \cdots - a_2(y^K) - a_1 \ . \end{array}$$

Store the lagrange multipliers ν^K. Update the upper bound so that

$$P^{UBD} = MIN(P^{UBD}, F(y^K)).$$

STEP 2- *Determination of nearest points from the previous values of* y*:*

Set $y^{LEFT} = y^L, y^{RIGHT} = y^U$. Set $LRD = YES$, $RRD = YES$. If $y^K = y^L$, then set $LRD = NO$. If $y^K = y^U$, then set $RRD = NO$.

(a) If $K = 1$, then set $y^{LEFT} = y^L$, $y^{RIGHT} = y^U$. Go to Step 3.

(b) If $K = 2$, then y^K is either the lower bound y^L or the upper bound y^U.

(i) If $y^2 = y^L$, then set $y^{LEFT} = y^L$, $y^{RIGHT} = y^1$. Go to Step 3.

(ii) If $y^2 = y^U$, then set $y^{LEFT} = y^1$, $y^{RIGHT} = y^U$. Set Go to Step 3.

(c) If $K > 2$, then then do the following steps for $k = 1, 2, \ldots, K-1$.

(i) If $y^k < y^K$, then set $y^{LEFT} = MAX(y^k, y^{LEFT}),\ L = k$

(ii) If $y^k > y^K$, then set $y^{RIGHT} = MIN(y^k, y^{RIGHT}),\ U = k$

STEP 3– *First Relaxed dual problem (i.e. for $y^K - y \geq 0$) :*

Updating bounds on x:

Reset the bounds on the x variables as follows :

$$\begin{array}{rclrcl} x_1^L & = & y^{LEFT}\ , & x_1^U & = & y^K \\ x_2^L & = & (y^{LEFT})^2\ , & x_2^U & = & (y^K)^2 \\ x_3^L & = & (y^{LEFT})^3\ , & x_3^U & = & (y^K)^3 \\ \vdots & & \vdots & \vdots & & \vdots \\ x_N^L & = & (y^{LEFT})^N\ , & x_N^U & = & (y^K)^N\ . \end{array}$$

Formulating the Lagrange functions

(i) For the Lagrange function from the Lth iteration (i.e. the Lagrange function from the Lth constraint with the *qualifying* constraint $y^L - y \leq 0$) :

$$\left. \begin{array}{l} If \quad \nu_{i+1}^L \geq 0, \quad then \ \ x_i^{B_L} = x_i^U \\ If \quad \nu_{i+1}^L < 0, \quad then \ \ x_i^{B_L} = x_i^L \end{array} \right\} \quad \forall\ i = 1, 2, ..., N-1\ .$$

(ii) For the Lagrange function from the Kth iteration (i.e. the Lagrange function from the current iteration with the *qualifying* constraint $y^K - y \geq 0$) :

$$\left. \begin{array}{l} If \quad \nu_{i+1}^K \geq 0, \quad then \ \ x_i^{B_K} = x_i^L \\ If \quad \nu_{i+1}^K < 0, \quad then \ \ x_i^{B_K} = x_i^U \end{array} \right\} \quad \forall\ i = 1, 2, ..., N-1\ .$$

Solving the Relaxed dual problem

Find the intersection of the following two constraints :

$$\begin{array}{rcl} \mu_B & = & L(x^{B_L}, y, \nu^L), \text{ and} \\ \mu_B & = & L(x^{B_K}, y, \nu^K). \end{array}$$

If $\mu_B < P^{UBD}$, then store the solution of the problem. That is, set

$$\mu_B^{stor^1}(K) = \mu_B^{int}, \quad and \quad y^{stor^1}(K) = y^{int},$$

where *int* denotes the value of the variable obtained by the intersection.

STEP 4– *Second Relaxed dual problem: (i.e. for $y^K - y \leq 0$) :*

Updating bounds on x:

Reset the bounds on the x variables as follows :

$$\begin{array}{rclrcl} x_1^L & = & y^K \ , & x_1^U & = & y^{RIGHT} \\ x_2^L & = & (y^K)^2 \ , & x_2^U & = & (y^{RIGHT})^2 \\ x_3^L & = & (y^K)^3 \ , & x_3^U & = & (y^{RIGHT})^3 \\ \vdots & & \vdots & \vdots & & \vdots \\ x_N^L & = & (y^K)^N \ , & x_N^U & = & (y^{RIGHT})^N . \end{array}$$

Formulating the Lagrange functions

(i) For the Lagrange function from the Uth iteration :

$$\left.\begin{array}{l} If \quad \nu_{i+1}^U \geq 0, \quad then \ \ x_i^{B_U} = x_i^L . \\ If \quad \nu_{i+1}^L < 0, \quad then \ \ x_i^{B_U} = x_i^U . \end{array}\right\} \quad \forall \ i = 1, 2, ..., N-1 \ .$$

(ii) For the Lagrange function from the Kth iteration :

$$\left.\begin{array}{l} If \quad \nu_{i+1}^K \geq 0, \quad then \ \ x_i^{B_K} = x_i^U . \\ If \quad \nu_{i+1}^K < 0, \quad then \ \ x_i^{B_K} = x_i^L . \end{array}\right\} \quad \forall \ i = 1, 2, ..., N-1 \ .$$

Solving the Relaxed dual problem

Find the intersection of the following two constraints :

$$\begin{array}{rcl} \mu_B & = & L(x^{B_U}, y, \nu^U), \text{ and} \\ \mu_B & = & L(x^{B_K}, y, \nu^K). \end{array}$$

If $\mu_B int < P^{UBD}$, then store the solution of the problem. That is, set

$$\mu_B^{stor^2}(K) = \mu_B^{int}, \quad and \quad y^{stor^2}(K) = y^{int}.$$

where again *int* denotes the value of the variable obtained by the intersection.

STEP 5- *Selecting a new lower bound and y^{K+1}:*

From the stored sets $\mu_B^{stor^1}$ and $\mu_B^{stor^2}$, select the minimum μ_B^{min} (including the solutions from the current iteration). Also, select the corresponding stored value of y as y^{min}. Set $R^{LBD} = \mu_B^{min}$, and $y^{K+1} = y^{min}$. Delete μ_B^{min} and y^{min} from the stored set.

STEP 6- *Check for convergence:*

Check if $R^{LBD} > P^{UBD} - \epsilon$. IF yes , STOP. Else, set $K = K + 1$ and return to step 1.

Remark 1 : Since the primal problem in Step 1 is solved for a fixed value of y, the problem is just a function evaluation at $y = y^K$.

Remark 2 : Each of the two relaxed dual problems solved (in Steps 3 and 4) are solved by calculation of the intersection of two linear equality constraints in two variables and the comparison of the resulting solution for μ_B with the values of the objective function at the two relevant bounds of y.

5 An Illustrating Example

Consider the following unconstrained optimization problem :

$$\min_y \; -y^3 + 4.5y^2 - 6y$$

$$s.t. \qquad 0 \le y \le 3 \; .$$

Thus, $a_0 = 0, a_1 = -6, a_2 = 4.5$, and $a_3 = -1$. The nonconvexity in this problem arises due to the presence of the term $-y^3$ in the objective function.

Introduction of three transformation variables x_1, x_2 and x_3 and their equivalence relationships to y results in the following equivalent form of the problem :

$$\min \; -6x_1 + 4.5x_2 - x_3$$

$$\begin{aligned} s.t. \quad x_1 - y &= 0 \\ x_2 - x_1 y &= 0 \\ x_3 - x_2 y &= 0 \\ 0 \le y &\le 3 \\ 0 \le x_1 &\le 3 \\ 0 \le x_2 &\le 9 \\ 0 \le x_3 &\le 27 \; . \end{aligned}$$

Consider a starting point of $y^1 = 1$ for the **GOP** algorithm. The solution of the primal problem can be found as follows :

$$\begin{array}{rcccl} x_0 & = & 1 & & \\ x_1 & = & (y^1) & = & 1 \\ x_2 & = & (y^1)^2 & = & 1 \\ x_3 & = & (y^1)^3 & = & 1 \\ F(y^1) & = & \sum_{i=0}^{3} a_i (y^1)^i & = & -2.5 \\ \nu_3^1 & = & -a_3 & = & 1 \\ \nu_2^1 & = & \nu_3^1 y^1 - a_2 & = & -3.5 \\ \nu_1^1 & = & \nu_2^1 y^1 - a_1 & = & 2.5 \ . \end{array}$$

Since $K = 1$, and $y^1 = 1$, set $y^{LEFT} = y^L = 0$ and $y^{RIGHT} = y^U = 3$. Also, set $LRD = YES$ and $RRD = YES$.

The Lagrange function for the problem can be formulated as

$$L(x, y, \nu^1) = -3.5x_1(1 - y) + x_2(1 - y) - 2.5y \ .$$

<u>First Relaxed dual problem</u> (Solved for $1 - y \geq 0$)

According to Step 3 of the modified algorithm, set :

$$\begin{array}{rcccl} x_1^L & = & y^{LEFT} & = & 0\ , \\ x_2^L & = & (y^{LEFT})^2 & = & 0\ , \\ x_3^L & = & (y^{LEFT})^3 & = & 0\ , \end{array} \qquad \begin{array}{rcccl} x_1^U & = & y^1 & = & 1 \\ x_2^U & = & (y^1)^2 & = & 1 \\ x_3^U & = & (y^1)^3 & = & 1\ . \end{array}$$

The proper bounds to be used for the x variables in the Lagrange function are then determined as follows :

$$\begin{array}{ll} Since & \nu_2^1 < 0, \ \ x_1^{B_1} = x_1^U = 1. \\ Since & \nu_3^1 > 0, \ \ x_2^{B_1} = x_2^L = 0. \end{array}$$

Using these bounds for the x variables, the current Lagrange function becomes

$$L(x, y, \nu^1) = -3.5(1 - y) - 2.5y = y - 3.5.$$

Since this is the first iteration, there are no previous Lagrange functions. Therefore, the solution to this problem will lie at $y = 0$, since the coefficient of y in the Lagrange function is positive. The objective value, that is, the value of μ_B, is -3.5 .

<u>Second Relaxed dual problem</u> (Solved for $1 - y \leq 0$)

Set

$$
\begin{array}{rclclrclcl}
x_1^L &=& y^1 &=& 1\ , & x_1^U &=& y^{RIGHT} &=& 1, \\
x_2^L &=& (y^1)^2 &=& 1\ , & x_2^U &=& (y^{RIGHT})^2 &=& 9, \\
x_3^L &=& (y^1)^3 &=& 1\ , & x_3^U &=& (y^{RIGHT})^3 &=& 27.
\end{array}
$$

The proper bounds to be used for the x variables in the Lagrange function are then determined as follows :

$$
\begin{array}{ll}
\textit{Since} & \nu_2^1 < 0, \quad x_1^{B_1} = x_1^L = 1. \\
\textit{Since} & \nu_3^1 > 0, \quad x_2^{B_1} = x_2^U = 9.
\end{array}
$$

Using these bounds for the x variables, the current Lagrange function becomes

$$L(x, y, \nu^1) = -3.5(1 - y) + 9(1 - y) - 2.5y = -8y + 5.5.$$

The solution can be found by inspection. In this case, since the coefficient of y is negative, the solution lies at the upper bound, that is, $y = 3, \mu_B = -18.5$.

Thus, at the end of the first iteration, the lower bound from the two relaxed dual problems is -18.5 . The value of y for the next iteration is 3.

It is interesting to compare the two solutions found in the first iteration to the solutions that would be found using the original bounds for the x variables. If the bounds for x_1, x_2 and x_3 were fixed to be $[0, 3]$, $[0, 9]$ and $[0, 27]$ (which are the original bounds for the x variables), then the solutions found are $y = 0, \mu_B = -10.5$ and $y = 3, \mu_B = -25.5$. Thus, using the improved bounds for x results in a tighter lower bound from the relaxed dual problems.

According to Step 5, the next y is selected as the one that corresponds to the minimum of the two relaxed dual problems, that is, $y^2 = 3$.

<u>Second Iteration</u>

The primal problem is solved for $y^2 = 3$. The solution to this problem is:

$$
\begin{array}{rclcl}
x_0 &=& 1 && \\
x_1 &=& y^2 &=& 3 \\
x_2 &=& (y^2)^2 &=& 9 \\
x_3 &=& (y^2)^3 &=& 27 \\
F(y^2) &=& \sum_{i=0}^{3} a_i (y^1)^i &=& -4.5 \\
\nu_3^2 &=& -a_3 &=& 1 \\
\nu_2^2 &=& \nu_3^2 y^2 - a_2 &=& -1.5 \\
\nu_1^2 &=& \nu_2^2 y^2 - a_1 &=& 1.5.
\end{array}
$$

Since $K = 2$, and $y^1 = 1 < y^2$, set $y^{LEFT} = y^1 = 1$ and $y^{RIGHT} = y^U = 3$. Also, since $y^2 = y^U$, set $LRD = YES$ and $RRD = NO$.

The Lagrange function for the problem can be formulated as

$$L(x, y, \nu^2) = -1.5x_1(3 - y) + x_2(3 - y) - 1.5y.$$

Since $RRD = NO$, only the relaxed dual problem for $y \leq 3$ needs to be solved.

Relaxed dual problem (Solved for $3 - y \geq 0$)

Set

$$\begin{array}{lllllllll} x_1^L & = & y^{LEFT} & = & 1\,, & x_1^U & = & y^1 & = & 3, \\ x_2^L & = & \left(y^{LEFT}\right)^2 & = & 1\,, & x_2^U & = & \left(y^1\right)^2 & = & 9, \\ x_3^L & = & \left(y^{LEFT}\right)^3 & = & 1\,, & x_3^U & = & \left(y^1\right)^3 & = & 27. \end{array}$$

The proper bounds to be used for the x variables in the Lagrange function from the current iteration are then determined as follows :

$$\begin{array}{ll} Since & \nu_2^2 < 0, \;\; x_1^{B_2} = x_1^U = 3. \\ Since & \nu_3^2 > 0, \;\; x_2^{B_2} = x_2^L = 1. \end{array}$$

Using these bounds for the x variables, the current Lagrange function becomes

$$L(x, y, \nu^2) = -4.5(3 - y) + 1(3 - y) - 1.5y = 2y - 10.5.$$

For the previous iteration, the Lagrange function has the following bounds for x:

$$\begin{array}{ll} Since & \nu_3^2 > 0, \;\; x_2^{B_1} = x_2^U = 9. \\ Since & \nu_2^2 < 0, \;\; x_1^{B_1} = x_1^L = 1. \end{array}$$

From this, the Lagrange function from the first iteration becomes

$$L(x, y, \nu^1) = -3.5(1 - y) + 9(1 - y) - 2.5y = -8y + 5.5.$$

The solution to the current relaxed dual problem lies at the intersection of the two Lagrange functions, or at one of the two bounds for y in the current problem, that is, at either $y = 1$ or $y = 3$. If the solution lies at either of these two bounds, then it need not be considered. Therefore, the only case to be considered is when the solution lies at the intersection of the Lagrange functions. Here, the intersection lies at $y = 1.6, \mu_B = -7.3$. This is less than the objective function values at $y = 1$ or $y = 3$, so this is the solution of the current relaxed dual problem. Therefore, $R^{LBD} = -7.3$ and $y^3 = 1.6$.

Third Iteration

The primal problem is solved for $y^3 = 1.6$. The solution to this problem is:

$$\begin{array}{rcccl} x_0 & = & 1 & & \\ x_1 & = & y^3 & = & 1.6 \\ x_2 & = & (y^3)^2 & = & 2.56 \\ x_3 & = & (y^3)^3 & = & 4.096 \\ F(y^3) & = & \sum_{i=0}^{3} a_i (y^3)^i & = & -2.176 \\ \nu_3^3 & = & -a_3 & = & 1 \\ \nu_2^3 & = & \nu_3^3 y^3 - a_2 & = & -2.9 \\ \nu_1^3 & = & \nu_2^3 y^3 - a_1 & = & 1.36. \end{array}$$

Since $y^1 = 1 < y^3$, and $y^2 = 3 > y^3$, set $y^{LEFT} = y^1 = 1$ and $y^{RIGHT} = y^2 = 3$. Also, set $LRD = YES$ and $RRD = YES$.

The Lagrange function for the problem becomes

$$L(x, y, \nu^3) = -2.9x_1(1.6 - y) + x_2(1.6 - y) - 1.36y.$$

First Relaxed dual problem (Solved for $1.6 - y \geq 0$)

Set

$$\begin{array}{rclcl} x_1^L & = & y^{LEFT} & = & 1\,, \\ x_2^L & = & (y^{LEFT})^2 & = & 1\,, \\ x_3^L & = & (y^{LEFT})^3 & = & 1\,, \end{array} \qquad \begin{array}{rclcl} x_1^U & = & y^3 & = & 1.6, \\ x_2^U & = & (y^3)^2 & = & 2.56, \\ x_3^U & = & (y^3)^3 & = & 4.096. \end{array}$$

The proper bounds to be used for the x variables in the Lagrange function are then determined as follows :

$$\begin{array}{l} \textit{Since} \quad \nu_3^3 > 0, \quad x_2^{B_3} = x_2^L = 1. \\ \textit{Since} \quad \nu_2^3 < 0, \quad x_1^{B_3} = x_1^U = 1.6. \end{array}$$

Using these bounds for the x variables, the current Lagrange function becomes

$$L(x, y, \nu^3) = 2.28y - 5.824.$$

Since $y^1 = 1 < y^3$ and the current relaxed dual problem is being solved for $y \leq 1.6$, the Lagrange function from the first iteration for $y \geq 1$ will be present. With the above bounds for x_1 and x_2, this Lagrange function becomes

$$L(x, y, \nu^1) = -3.5(1 - y) + 2.56(1 - y) - 2.5y = -1.56y - 0.94.$$

The intersection of the two Lagrange functions lies at $y = 1.2718$ and $\mu_B = -2.924$. Since the value of μ_B is greater than the best upper bound from the primal problems (-4.5), this solution need not be considered for future iterations.

Second Relaxed dual problem (Solved for $1.6 - y \leq 0$)

Set

$$\begin{array}{rclclcrclcl} x_1^L & = & y^3 & = & 1.6\ , & & x_1^U & = & y^{RIGHT} & = & 3, \\ x_2^L & = & (y^3)^2 & = & 2.56\ , & & x_2^U & = & (y^{RIGHT})^2 & = & 9, \\ x_3^L & = & (y^3)^3 & = & 4.096\ , & & x_3^U & = & (y^{RIGHT})^3 & = & 27. \end{array}$$

The proper bounds to be used for the x variables in the Lagrange function are then determined as follows :

$$\begin{array}{l} \textit{Since} \quad \nu_2^3 < 0, \;\; x_1^{B_3} = x_1^L = 1.6. \\ \textit{Since} \quad \nu_3^3 > 0, \;\; x_2^{B_3} = x_2^U = 9. \end{array}$$

Using these bounds for the x variables, the current Lagrange function becomes

$$L(x, y, \nu^3) = 6.976 - 5.72y.$$

Since $y^2 = 3 > y^3$ and the current relaxed dual problem is being solved for $y \geq 1.6$, the Lagrange function from the second iteration for $y \leq 3$ will be present. With the above bounds for x_1 and x_2, this Lagrange function becomes

$$L(x, y, \nu^2) = -4.5(3 - y) + 2.56(3 - y) - 1.5y = 0.44y - 5.82.$$

The solution to this relaxed dual problem lies at the intersection of the two Lagrange functions, which occurs at $y = 2.0772$ and $\mu_B = -4.9059$.

At the fourth iteration, the primal problem has a solution of -2.009. When the two relaxed dual problems are solved, they both have solutions greater than -4.5. Therefore, the algorithm converges at the end of the fourth iteration. In comparison to this, the original **GOP** algorithm takes 17 iterations to converge to the global solution.

6 Geometrical Interpretation

The application of the modified **GOP** algorithm to the example in the previous section can also be illustrated geometrically. Figure 1(a) shows the plot of the objective function $F(y)$ as a function of y. Since the problem is one of unconstrained minimization, is also the plot of the solutions of the primal problem as a function of y.

For a starting point of $y^1 = 1$, the sequence of points generated by the algorithm is graphically illustrated in Figures 1(b)-1(e). For the first iteration (Figure 1(b)), with an optimal value of -2.5 for the primal problem, the relaxed dual problems are solved for $y \leq 1$ and for $y \geq 1$. For the relaxed dual problem corresponding to $y \leq 1$, the bounds for x_1 and x_2 (which originally were [0,3] and [0,9] respectively) can be improved to [0,1] and [0,1] respectively. L_1^1 is the Lagrange function that results from using the improved bounds for the x variables, while P_1^1 is the Lagrange function obtained from using the original bounds for x_1 and x_2. Thus, the use of the improved bounds results in a tighter underestimator for the relaxed dual problem. Similarly, for the relaxed dual problem corresponding to $y \geq 1$, the bounds for x_1 and x_2 can be improved to [1,3] and [1,9] (instead of [0,3] and [0,9]). This results in a tighter Lagrange function (L_2^1) when the modified bounds are used as compared to the Lagrange function (P_2^1) that results from the use of the original bounds.

For the second iteration, $y^2 = 3$ for the primal problem, and the optimal solution is -4.5. For this iteration, only one relaxed dual problem needs to be solved, namely for $y \leq 3$. For this problem, the Lagrange function from the first iteration corresponding to $y \geq 1$ is present. Therefore, the bounds on x_1 and x_2 can be modified to [1,3] and [1,9] respectively. The Lagrange function that results from the use of the tighter bounds on the x variables, namely L_1^2, is a tighter underestimator of the objective function than the Lagrange function P_1^2 obtained from using the original bounds for x_1 and x_2. The solution of this relaxed dual problem is shown by point B on Figure 1(c). In contrast, the solution obtained by the original **GOP** algorithm at a similar juncture is shown by point A. As can be seen, the lower bound obtained from the modified algorithm is tighter than the lower bound from the original **GOP** algorithm. It should be noted that the original **GOP** algorithm and the modified **GOP** algorithm differ in the subsequent selections of y for the third and further iterations.

Figure 1(d) shows the relaxed dual problems solved at the third iteration, for which the corresponding primal problem has been soved for $y = 1.6$. For this iteration, the nearest points from previous iterations on the left and right sides are $y = 1$ and $y = 3$. Consider the relaxed dual problem solved for $y \leq 1.6$. The bounds on x_1 and x_2 can be improved to [1,1.6] and [1,2.56] respectively. This results in the Lagrange function L_1^3 formulated from the current iteration. It is interesting to note that due to the improvement of the bounds on the x variables results in an even tighter form of the Lagrange function from the first iteration. Originally, at the second relaxed dual problem in the first iteration, this Lagrange function had been formulated using the bounds of [1,3] and [1,9] for x_1 and x_2 respectively, and is represented by L_2^1 in Figure 1(d). Now, how-

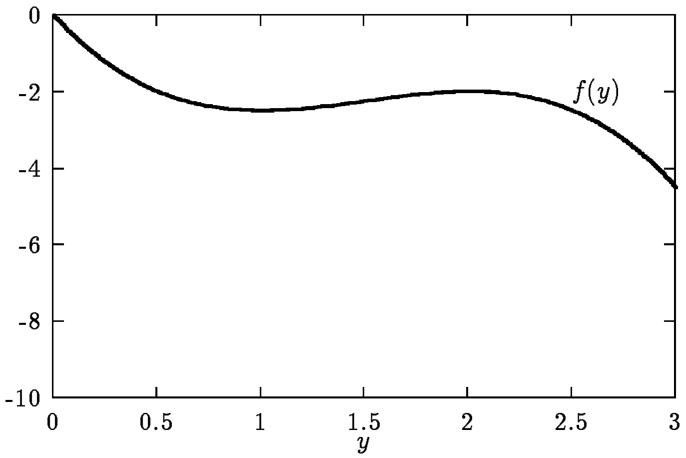

Figure 1(a) : Objective Function

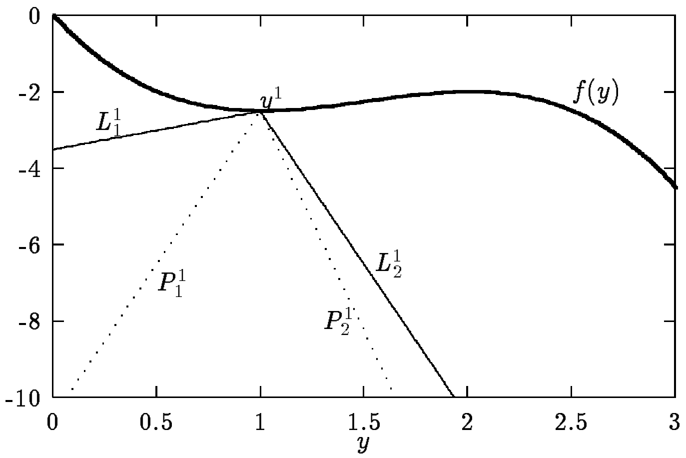

Figure 1(b) : Iteration 1 of the modified **GOP** algorithm

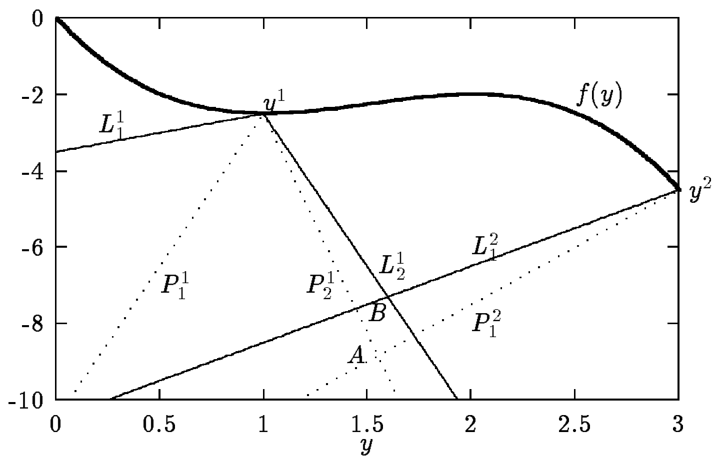

Figure 1(c) : Iteration 2 of the modified **GOP** algorithm

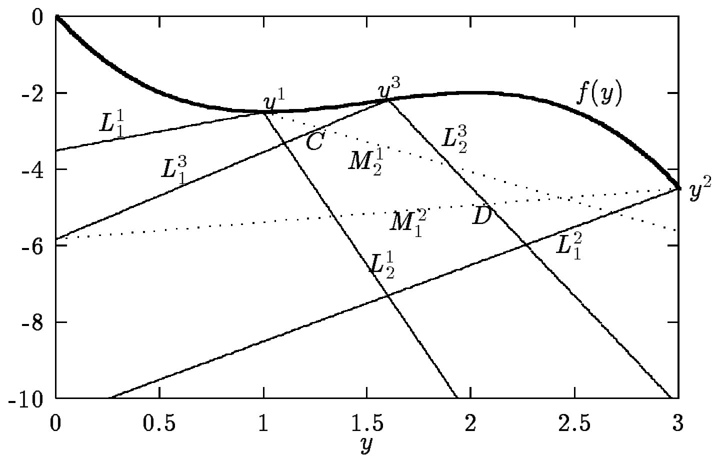

Figure 1(d) : Iteration 3 of the modified **GOP** algorithm

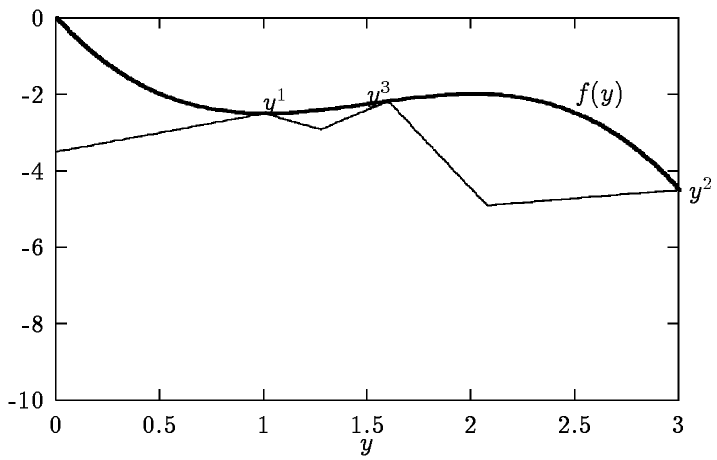

Figure 1(e) : Underestimating Function After 3 Iterations

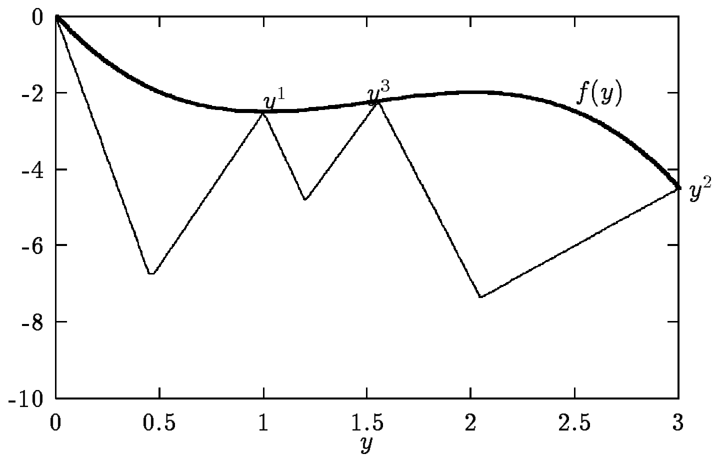

Figure 1(f) : Underestimating Function after 3 Iterations of the Original **GOP** Algorithm ever, the bounds on x_1 and x_2 are [1,1.6] and [1,2.56] respectively. Using these bounds results in the Lagrange function M_2^1 from the first iteration. Therefore, the solution of the current relaxed dual problem lies at the point C. Similarly, for the relaxed dual problem solved for $y \geq 1.6$, the bounds on x_1 and x_2 can be improved to [1.6,3] and [2.56,9] respectively. This results in the Lagrange function L_2^3 from the current iteration. At the same time, the Lagrange function from the second iteration moves up from L_1^2 to M_1^2 due to the use of the tighter bounds on the x variables. The solution of this relaxed dual problem is shown by the point D on Figure 1(d).

Figure 1(e) shows the underestimating function that is obtained after three iterations of the modified **GOP** algorithm. Figure 1(f) shows the underestimating function obtained after three iterations of the original **GOP** algorithm when applied to this problem from a starting point of $y^1 = 1$. As can be seen, the modified algorithm provides a tighter underestimator as compared to the the one that is obtained by the original algorithm. Moreover, the modified algorithm eliminates whole regions of the problem for future iterations due to the tightness of the underestimator.

At the fourth iteration, the Lagrange functions from the second and third iteration move up so that the solution of the relaxed dual problems is -4.5, and the algorithm terminates.

7 Constrained Problems

When the optimization problems involve constraints with polynomial functions in the objective function and/or constraints, the primal problem can no longer be solved by function evaluation. In this case, the primal problem, for a fixed $y = y^K$, can be written as

$$\min_x \sum_{i=0}^{N} a_i x_i$$

$$\begin{aligned} \sum_{i=0}^{N} A_{ji} x_i &\leq 0 \quad j = 1, 2, ..., J \\ \sum_{i=0}^{N} B_{mi} x_i &= 0 \quad m = 1, 2, ..., M \\ x_i - x_{i-1} y^K &= 0 \quad i = 1, 2, ..., N, \end{aligned}$$

where $x_0 = 1$.

The solution of this problem is obtained by solving a linear programming problem in the x variables, and provides the multipliers used in formulating the Lagrange function as well as an upper bound on the global solution.

The KKT gradient conditions for this problem are

$$\begin{aligned}
\nabla_{x_i} L(x, y^K, \lambda^K, \mu^K, \nu^K) &= a_i + \sum_{j=1}^{J} \mu_j^K A_{ji} + \sum_{m=1}^{M} \lambda_m^K B_{mi} \\
&\quad + \nu_i^K - \nu_{i+1}^K y^K \\
&= 0
\end{aligned}$$

where λ^K and μ^K correspond to the original equality and inequality constraints, and ν^K corresponds to the new equality constraints introduced, with $\nu_0^K = \nu_{N+1}^K = 0$.

The Lagrange function for this problem is given by

$$\begin{aligned}
L(x, y, \lambda^K, \mu^K, \nu^K) &= \sum_{i=0}^{N} [a_i x_i + \sum_{j=1}^{J} \mu_j^K A_{ji} x_i \\
&\quad + \sum_{m=1}^{M} \lambda_m^K B_{mi} x_i + \nu_i^K (x_i - x_{i-1} y)].
\end{aligned}$$

Separating the terms in x, this can also be written as

$$\begin{aligned}
L(x, y, \lambda^K, \mu^K, \nu^K) &= \sum_{i=0}^{N} \Bigg[a_i + \sum_{j=1}^{J} A_{ji} \mu_j^K + \sum_{m=1}^{M} B_{mi} \lambda_m^K \\
&\quad + \nu_i^K - \nu_{i+1}^K y \Bigg] x_i.
\end{aligned}$$

Using the KKT conditions, the Lagrange function can be written as

$$L(x, y, \lambda^K, \mu^K, \nu^K) = \sum_{i=0}^{N} \nu_{i+1}^K (y^K - y) x_i.$$

Thus, it can be seen that the Lagrange function formulated from the primal problem is identical to the one formulated in the case of unconstrained optimization problems. Therefore, Steps 2-6 of the improved **GOP** algorithm presented in Section 4 remain the same as for unconstrained optimization problems.

It is possible that for some values of $y = y^K$, the primal problem is infeasible. In this case, it is necessary to solve a relaxed primal problem involving the minimization of the sum of infeasibilities. This problem is shown below:

$$\min_x \sum_{j=1}^{J} \alpha_j + \sum_{m=1}^{M} (\beta_m + \gamma_m)$$

$$\begin{aligned}
\sum_{i=0}^{N} A_{ji} x_i &\leq \alpha_j \quad j = 1, 2, ..., J \\
\sum_{i=0}^{N} B_{mi} x_i &= \beta_m - \gamma_m \quad m = 1, 2, ..., M \\
x_i - x_{i-1} y^K &= 0 \quad i = 1, 2, ..., N \ .
\end{aligned}$$

The Lagrange function for this problem is given by

$$\begin{aligned}
L(x, y, \lambda^K, \mu^K, \nu^K) &= \sum_{j=1}^{J} \alpha_j + \sum_{m=1}^{M} (\beta_m + \gamma_m) + \sum_{j=1}^{J} \mu_j^K \sum_{i=1}^{N} A_{ji} x_i - \alpha_j \\
&+ \sum_{m=1}^{M} \lambda_m^K (\sum_{i=1}^{N} B_{mi} x_i - \beta_m + \gamma_m) \\
&+ \sum_{i=1}^{N} \nu_i^K (x_i - x_{i-1} y) \ .
\end{aligned}$$

Using the KKT gradient conditions for the relaxed primal problem and separating the terms in x, this again be reduces to

$$L(x, y, \lambda^K, \mu^K, \nu^K) = \sum_{i=0}^{N} \nu_{i+1}^K (y^K - y) x_i.$$

In this case, however, the Lagrange function is added to the relaxed dual problem in the following form:

$$0 \geq L(x, y, \lambda^K, \mu^K, \nu^K).$$

This has the same form as the regular Lagrange function, except that μ_B has been replaced by 0. Steps 2-6 can again be applied towards solving relaxed dual problems in these iterations, with the above replacement being the only change.

8 Computational Experience

8.1 Unconstrained Problems

Example 1 :

This example is taken from Wingo (1985).

$$\min_y \; y^6 - \frac{52}{25}y^5 + \frac{39}{80}y^4 + \frac{71}{10}y^3 - \frac{79}{20}y^2 - y + \frac{1}{10}$$

$$-2 \le y \le 11$$

This function has a local minimum at 0, with a value of $\frac{1}{10}$. The best solution reported by Wingo (1985) is -23627.1758, occurring at $y = 11$. However, the global minimum of the function occurs at $y = 10$, with an objective value of -29763.233 .

For this problem,

$$N = 6, \quad a = (\frac{1}{10}, -1, -\frac{79}{20}, \frac{71}{10}, \frac{39}{80}, -\frac{52}{25}, 1), \quad M = J = 0.$$

When the **(GOP)** was applied to this problem, it converged to the global solution of -29763.233 taking around 175 iterations from different starting points. When the improved **GOP** algorithm was applied, however, the global optimum was identified from all starting points in less than 24 iterations. For a relative tolerance of 10^{-3}, the algorithm takes 11 iterations , while the number of iterations required for relative error between the upper and lower bounds to be less than 10^{-8} is 23. After 26 iterations, the algorithm converges exactly (with an absolute error of 0) to the global solution.

Example 2 :

This example is taken from Moore (1979). It involves the minimization of a 50^{th} degree polynomial in one variable.

$$\min_y \; \sum_{i=1}^{50} a_i y^i$$

$$1 \le y \le 2$$

where

$$\begin{aligned} a \;=\; \{&-500.00000, 2.5000000, 1.666666666, 1.2500000, 1.000000, \\ &0.8333333, 0.714285714, 0.625000000, 0.555555555, 1.0000000, \\ &-43.6363636, 0.41666666, 0.384615384, 0.357142857, 0.3333333, \\ &0.312500000, 0.294117647, 0.277777777, 0.263157894, 0.2500000, \\ &0.238095238, 0.227272727, 0.217391304, 0.208333333, 0.2000000, \\ &0.192307692, 0.185185185, 0.178571428, 0.344827586, 0.6666666, \\ &-15.483870970, 0.15625000, 0.1515151, 0.14705882, 0.14285712,\end{aligned}$$

$$0.138888888, 0.135135135, 0.131578947, 0.128205128, 0.1250000,$$
$$0.121951219, 0.119047619, 0.116279069, 0.113636363, 0.1111111,$$
$$0.108695652, 0.106382978, 0.208333333, 0.408163265, 0.8000000\}.$$

This function has the global minimum at $y = 1.0911$, with a value of -663.5. The improved version of the **GOP** algorithm takes 45 iterations to find the global solution from a starting point of $y = 1$.

Example 3 :

This example is taken from Wilkinson (1963).

$$\min_y \quad 0.000089248y - 0.0218343y^2 + 0.998266y^3 - 1.6995y^4 + 0.2y^5$$

$$0 \le y \le 10.$$

This problem has local minima at $y = 6.325, f = -443.67$, $y = 0.4573$, $f = -0.02062$, $y = 0.01256, f = 0.0$ and $y = 0.00246, f = 0$ among others. The improved **GOP** algorithm identifes the global solution from different starting points within 25 iterations.

Example 4 :

This example is taken from Dixon and Szego (1975).

$$\min_y \quad 4y^2 - 4y^3 + y^4$$

$$-5 \le y \le 5.$$

This problem has two global minima at $y = 0, f = 0$, and $y = 2, f = 0$. There is a local maximum at $y = 1$. When the original **GOP** algorithm was applied to this problem, the global solutions were identified after around 150 iterations. However, the improved **GOP** algorithm identifes the global solution from different starting points within 50 iterations.

Example 5 (Three-hump camel-back function):

This example is taken from Dixon and Szego (1975).

$$\min \quad F(y) = 2y_1^2 - 1.05y_1^4 + \frac{1}{6}y_1^6 - y_1y_2 + y_2^2$$

$$-5 \le y_1, y_2 \le 5.$$

The nonconvexities in the problem are due to the $-1.05y_1^4$ and $-y_1y_2$ terms in the function. It can be seen that at the optimal solution, the value of y_2^* must be either -5, 5 or $\frac{y_1^*}{2}$. Assuming that the solution does not lie at -5 or 5, y_2 can be replaced by $\frac{y_1}{2}$. Then, the problem can be converted to the following unconstrained optimization problem :

$$\min \quad F(y) = 1.75y_1^2 - 1.05y_1^4 + \frac{1}{6}y_1^6$$
$$-5 \leq y_1 \leq 5.$$

The modified **GOP** algorithm was applied to the problem in this form. The solution of $y_1 = 0, F(y) = 0$ was identified in 31 iterations for an absolute error of 10^{-5} between the upper and lower bounds on the global solution.

Example 6 :

This example is taken from Goldstein and Price (1971).

$$\min \quad y^6 - 15y^4 + 27y^2 + 250$$
$$-5 \leq y \leq 5.$$

This function has local minima at (0,250), (3,7) and (-3,7). When the **GOP** algorithm was applied to the problem, the two global solutions were simultaneously identified in 68 iterations for a relative tolerance of 10^{-3}.

Example 7 :

This example is taken from Dixon (1990).

$$min \quad y^4 - 3y^3 - 1.5y^2 + 10y$$
$$-5 \leq y \leq 5.$$

This function has a global solution of -7.5 at $y = -1$. When the modified **GOP** algorithm was applied to the problem, the global solution was identified in 24 iterations for a relative tolerance of 10^{-3}.

Remarks :

The number of iterations taken by the above problems for different orders of accuracy in the convergence of the upper and lower bounds is given in Table 1. Some interesting points to note about the computational results of the modified algorithm are given below :

(a) For all the problems for which the algorithm was applied, the number of iterations required for convergence increases almost linearly with the accuracy desired. For Example 1, the number of iterations required for an accuracy of 10^{-3} is 11, while the number of iterations required for the bounds to be within 10^{-9} is only 26. This is in direct contrast to most algorithms that require an increasingly larger number of iterations as the accuracy is increased.

(b) For three of the problems considered (Examples 1,2 and 3), the algorithm terminates exactly after a certain number of iterations. This is because after some iterations, the use of the improved bounds for the x variable results in tighter underestimating functions. Eventually, at some iteration, all the points in the stored sets are used up, and the Lagrange functions formulated for that iteration are such that no new points are generated that can improve the solution. Thus, for these problems, the algorithm converges to the global solution exactly.

(c) For Examples 4 and 5, the convergence is in terms of the absolute difference between the upper and lower bounds. This is because the global solution of these problems is 0.

(d) At every iteration of the algorithm, there is one function evaluation associated with the solution of the primal problem, and two problems involving the solution of the simultaneous solution of two linear equations in two variables (which correspond to the solution of the two relaxed dual problems).

Problem	*Relative Tolerance for Convergence*			
	10^{-3}	10^{-5}	10^{-7}	10^{-8}
Example 1	11	15	19	23
Example 2	34	39	45	45
Example 3	13	18	28	33
Example 4	30	40	54	56
Example 5	27	31	34	36
Example 6	68	161	-	-
Example 7	24	77	216	491

Table 1: Number of iterations of the modified **GOP** algorithm

8.2 Constrained Problems

Example 8 :

This example is taken from Soland (1971).

$$\min_{y} \quad -12y_1 - 7y_2 + y_2^2$$

$$\begin{aligned} \textit{subject to} \quad -2y_1^4 + 2 - y_2 &= 0 \\ 0 \le y_1 &\le 2 \\ 0 \le y_2 &\le 3. \end{aligned}$$

The nonconvexity in this problem comes from the presence of the polynomial term $-2y_1^4$ in the first constraint.

When the original **GOP** algorithm was applied to the problem in this form, from a starting point of 0 for y_1, the algorithm converged to the global solution of -16.73889 at $y = (0.7175, 1.47)$ in 89 iterations, solving 3 subproblems at every iteration. When the improved **GOP** algorithm was applied from the same starting point, the global solution was identified in 14 iterations.

It should be noted that the constraint can be written in the following form :

$$y_2 = 2 - 2y_1^4.$$

Then, using this constraint to substitute for y_2 and utilizing the bounds on y_2, the problem can be converted into the following unconstrained optimization problem :

$$min \;\; 4y_1^8 + 6y_1^4 - 12y_1 - 10$$

$$0 \le y_1 \le 1.$$

When the modified **GOP** algorithm was applied to the problem in this form, the algorithm converges exactly to the global solution in 14 iterations. It should also be noted that in this form, the problem is convex, and therefore any conventional solver should be able to identify the global solution.

Example 9:

This is a test example that has a feasible region consisting of two disconnected sub-regions.

$$\begin{aligned} \min_{y} \quad & -y_1 - y_2 \\ y_2 &\le 2 + 2y_1^4 - 8y_1^3 + 8y_1^2 \\ y_2 &\le 4y_1^4 - 32y_1^3 + 88y_1^2 - 96y_1 + 36 \\ 0 &\le y_1 \le 3 \\ 0 &\le y_2 \le 4. \end{aligned}$$

The constraint region for this problem is given in Figure 2(a). As can be seen, there are two distinct regions where the problem is feasible. Because of this reason, if a conventional **NLP** solver were applied to this problem, it is highly unlikely that the solver would converge to the global solution at point C. Depending on the starting point, the solution will be one of the points A, B, or C.

From a starting point of 0 for y_1, the original **GOP** algorithm takes 210 iterations to converge to the global solution of -5.50796 (occurring at $y_1 = 2.3295$). When the improved **GOP** algorithm was applied to it, however, the algorithm took 24 iterations to converge to the global solution.

Conclusions

In this paper, the application of the **GOP** algorithm (Floudas and Visweswaran, 1990) to problems involving polynomial functions in one variable is considered. The problem is solved by decomposition into a series of primal and *relaxed dual* problems. The solution of the primal problem can be obtained by simple function evaluations. The relaxed dual problem can be solved through two subproblems, each of which is shown to reduce to a problem of finding the intersection of two linear constraints in two variables. The simplfied primal and relaxed dual problems result in a modified algorithm that is computationally very efficient. The application of the modified algorithm is shown through an illustrating example that details the improvement of the modified algorithm over the original **GOP** algorithm both numerically and geometrically. Several examples of unconstrained and constrained polynomial function problems are presented to highlight the efficiency of the new algorithm.

Acknowledgement

The authors gratefully acknowledge financial support from the National Science Foundation under Grant CBT-8857013, as well as support from Amoco Chemical Co., Tennessee Eastman Co., and Shell Development Co. .

References

[1] Dixon, L. C. W., On Finding the Global Minimum of a Function of One Variable, Presented at the SIAM National Meeting, Chicago (1990).

[2] Dixon, L. C. W., and Szegö, G. P., *Towards Global Optimization*, North Holland, Amsterdam (1975).

[3] Evtushenko, Y. G., Numerical Methods for Finding Global Extrema (Case of Nonuniform mesh), *USSR Computational Mathematics and Mathematical Physics*, **11**, 1390 (1971).

[4] Floudas, C.A. and Visweswaran, V., A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs -I. Theory, *Computers and Chemical Engineering*, **14**, 1397 (1990).

[5] Floudas, C.A., and Visweswaran, V., A Primal-Relaxed Dual Global Optimization Approach, *Submitted for Publication* (1991).

[6] Galperin, E.A., The Cubic Algorithm, *Journal of Mathematical Analysis and Applications*, **112**, 635 (1985).

[7] Galperin, E.A., The Beta-Algorithm, *Journal of Mathematical Analysis and Applications*, **126**, 455 (1987).

[8] Goldstein, A. A., and Price, J. F., On Descent From Local Minima, *Mathematics of Computation*, **25** , 569 (1971).

[9] Hansen, E.R., Global Optimization using Interval Analysis: The One-Dimensional Case, *Journal of Optimization Theory and Applications*, **29**, 331 (1979).

[10] Hansen, P., Jaumard, B., and Lu, S-H., Global Optimization of Univariate Lipschitz functions : I. Survey and Properties, *To appear in Mathematical Programming* (1991).

[11] Hansen, P., Jaumard, B., and Lu, S-H., Global Optimization of Univariate Lipschitz functions : II. New Algorithms and Computational Comparisons, *To appear in Mathematical Programming* (1991).

[12] Hansen, P., Lu, S-H., and Jaumard, B., Global Minimization of Univariate Functions by Sequential Polynomial Approximation, *International Journal of Computer Mathematics*, **28**, 183 (1989).

[13] Moore, R., Methods and Applications in Interval Analysis, *SIAM Studies in Applied Mathematics*, **SIAM**, Philadelphia (1979).

[14] Piyavskii, S.A., An Algorithm for finding the Absolute Extremum of a Function, *USSR Comput. Math. Phys.*, **12**, 57 (1972).

[15] Ratschek, H., and Rokne, J., *New Computer Methods for Global Optimization*, Halsted Press (1988).

[16] Shen, Z., and Zhu, Y., An Interval Version of Schubert's Iterative Method for the Localization of the Global Maximum, *Computing*, **38**, 275 (1987).

[17] Shirov, V. S., Search for the Global Extremum of a Polynomial on a Parallelopiped, *USSR Comput. Maths. Math. Phys.*, **25**, 105 (1985).

[18] Timonov, L.N., An Algorithm for Search of A Global Extremum, *Engineering Cybernetics*, **15**, 38 (1977).

[19] Visweswaran, V. and Floudas, C.A., A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs -II. Application of Theory and Test Problems, *Computers and Chemical Engineering*, **14**, 1419 (1990a).

[20] Visweswaran, V. and Floudas, C.A., New Properties and Computational Improvement of the GOP Algorithm For Problems With Quadratic Objective Function and Constraints, *Submitted for Publication* (1990b).

[21] Wilkinson, J. H., *Rounding Errors in Algebraic Processes*, Prentice-Hall, Engelwood Cliffs, N.J. (1963).

[22] Wingo, D.R., Globally Minimizing Polynomials Without Evaluating Derivatives, *International Journal of Computer Mathematics*, **17**, 287 (1985).